UNITED STATES PATENT APPLICATION

FOR

METHOD AND SYSTEM FOR PROVIDING A CIRCLE OF TRUST ON A NETWORK

Inventors:

BHATNAGAR, Bhavna
LUO, Ping
CHENG, Qingwen
BHAT, Shivaram
XU, Hong
SUN, Wei
RANGANATHAN, Aravindan

Prepared by:
WAGNER, MURABITO & HAO, LLP
Two North Market Street
Third Floor
San Jose, California 95113

METHOD AND SYSTEM FOR PROVIDING A CIRCLE OF TRUST ON A NETWORK

FIELD OF THE INVENTION

Embodiments of the present invention relate to network security, and more particularly to providing a circle of trust among a plurality of associated entities.

BACKGROUND OF THE INVENTION

Referring to Figure 1, a block diagram of a network according to the conventional art is shown. As depicted in Figure 1, a plurality of organizations 105, 110, 115 are communicatively coupled by one or more communication channels 180 185, such as the internet or extranet. Each organization 105, 110, 115 typically comprises a plurality of client devices 120-150 communicatively coupled to one or more servers 155-175. The servers 155-175 provide one or more resources, such as execution of applications and/or storage of information.

A user on a client device 120-150 may be granted or denied access to resources of a particular server 155-175. In the conventional art, the client 120 logs-on to a particular organization's server 155, wherein the user provides a user name, password and/or the like. Based upon the user name, password and the like, the server 155 authenticates the client 120 and determines the client's 120 authorization to access particular resources.

If the client 120 then tries to access resource on another server 170, 175, establishing authentication and utilizing resources is problematic. The other servers 170, 175 do not know that the client device has been authenticated by a particular server 155. The other servers 170, 175 do not know that they can trust the authentication provided by the particular server 155. Furthermore, each entity 105, 110, 115 and/or server 155-175 may have a different login script, may require a different protocol, may store information in a different structure, format and/or the like. Therefore, the client typically has to sign-on to each server 155-175 separately.

For example, a user may wish to access resources on various entities 105-115 during the course of their work, such as using the internet to make travel arrangements. The user first logs-on to the company's network server 155 utilizing a client device 125. The user may manually or via a script, enter their user name and password in order to logon to the network server 155. The network server 155 provides an internet portal.

The user may then navigate using a browser to the website of an airline 115. The user will likely be required to enter a user name, password and the like to book a flight using a corporate account. Similarly, the user may then navigate to a car rental agency to reserve a rental car. Once again, the user may be requested to enter a name, password and the like to reserve the car. Similarly, the user may also navigate to a website of a hotel chain to reserve a room. Once again, the user may be requested to enter a name, password and the like to reserve the room. The need to logon to each entity's server 155 reduces the user's satisfaction and productivity.

The need to logon multiple times is not limited to multiple entities' servers 170, 175. For example, the user may login to their employer's network server 155 to access the finance server 160. The user may again be required to enter a username, password and the like in order to enter expenses, such as meals, entertainment and gas, incurred during their business travel. The user may then wish to check their retirement account. Once again the user may be required to provide a username, password and the like to access the payroll server 165 in order to check their retirement account. In addition to reducing the user's satisfaction and productivity, the implementation of multiple logon scripts increases the cost of doing business.

Versions of single sign-on services have existed for several years. However, the conventional art single sign-on services are closed solutions that do not offer broad interoperability. Accordingly, the Security Assertion Markup Language (SAML) specification is intended to provide a solution allowing single sign-on for secure authentication and authorization.

SAML is an eXtensible Markup Language (XML) standard designed for business-to-business (B2B) and business-to-consumer (B2C) transactions. The SAML standard is designed for the exchange of secure sign-on information between a user, a relying party, and/or an issuing party. Furthermore, SAML allows issuing parties to use their own chosen methods of authentication (e.g., personal key identifier (PKI), hash, password, or the like).

In one implementation, a SAML-compliant service, called a relying party, sends a SAML request to an issuing party, which returns a SAML assertion. Assertions do not create a secure authentication. The security service is responsible for providing a secure authentication. Assertions are coded statements generated about events, such as authentication, that have already occurred, as when the user provided the correct user name and password, or the security mechanism granted specific permissions.

Referring now to Figure 2A, an exemplary SAML request/assertion according to the conventional art is shown. As depicted in Figure 2A, SAML requests and assertions 210 are transmitted within a SOAP envelope 215 via HTTP 220.

Referring now to Figure 2B, an exemplary SAML data packet according to the conventional art is shown. As depicted in Figure 2B, the data packet comprises an HTTP header 250, a SOAP header 255 and a SAML payload 260. An assertion or response is encoded into the SAML payload 260. A SOAP header 255 is then generated and attached to the SAML payload 260. An HTTP header 250 is then generated and attached to the SOAP header 255 and SAML payload 260. The SAML payload containing an assertion or request may comprise an issuer identifier, an assertion identifier, an optional subject, an optional advice, a condition, an audience restriction, a target restriction, and an application specific condition.

Upon receipt, the HTTP header 250 is processed to provide routing and flow control. The SOAP header 255 is then processed to provide information concerning the

content of the payload and how to process it. The SAML payload 260 may then be processed to provide security information.

SUMMARY OF THE INVENTION

Embodiments of the present invention provide a circle of trust on a network. The circle of trust provides affiliated entities a means for determining that a user has been authenticated. The circle of trust also provides the affiliated entities a means for determining whether they can trust the authentication provided by another entity. Embodiments of the present invention provide for configuring a plurality of affiliated entities to establish a circle of trust. Embodiments of the present invention also provide for a circle of trust session.

In one embodiment of the present invention, the method of providing a circle of trust comprises exchanging a first certificate of a first affiliated and a second affiliated entity. The credential of the first affiliated entity is stored in a trusted partner list of the second affiliated entity. The credential of the second affiliated entity is stored in a trusted partner list of the first affiliated entity. Thereafter, a circle of trust session may be provided when a client device initiates use of a resource on a relying party device by providing an authentication assertion reference. The identity of the issuing party of the authentication is determined as a function of the authentication assertion reference. The relying party sends an authentication request containing its credential to the issuing party The issuing party determines if the relying party is a trusted entity based upon whether the relying party's credential is contained in the trusted partner list of the issuing party.

In another embodiment of the present invention, the circle of trust comprises a plurality of affiliated entities. Each affiliated entity in a circle of trust comprises an

administration module, a trusted partner list, a session module and/or an authentication module. The administration modules provide for the exchange of credentials between each affiliated entity. The administration modules also cause the credentials to be saved on each of the corresponding trusted partner lists. The session modules provide for generating and processing authentication requests and assertions. The session modules also provide for determining the identity of an issuing party as a function of an authentication assertion reference. The session modules also provide for determining a trusted status of a particular affiliated entity as a function of a certificate received from a relying party.

Accordingly, embodiments of the present invention provide single sign-on for secure authentication and authorization. Embodiments of the present invention enable open and interoperable designs for web-based single sign-on service functionality. Furthermore, embodiments of the present invention allow each affiliated entity to use their own chosen methods of authentication (e.g., personal key identifier (PKI), hash, password, or the like). Hence, the above-described advantageous provided by embodiments of the present invention increase user satisfaction and productivity.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example and not by way of limitation, in the figures of the accompanying drawings and in which like reference numerals refer to similar elements and in which:

Figure 1 shows a block diagram of a network according to the conventional art.

Figure 2A shows an exemplary SAML request/assertion according to the conventional art.

Figure 2B shows an exemplary SAML data packet according to the conventional art.

Figure 3 shows a flow diagram of a computer implemented method of configuring a circle of trust between a first and second affiliated entity, in accordance with one embodiment of the present invention.

Figure 4 shows an exemplary trusted partner list, in accordance with one embodiment of the present invention.

Figure 5 shows a block diagram of a system providing for configuring a circle of trust, in accordance with one embodiment of the present invention.

Figure 6 shows a flow diagram of a computer implemented method of providing a circle of trust session between a first and second affiliated entity, in accordance with one embodiment of the present invention.

Figure 7 shows a block diagram of a system providing for a circle of trust session, in accordance with one embodiment of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

Reference will now be made in detail to the embodiments of the invention, examples of which are illustrated in the accompanying drawings. While the invention will be described in conjunction with these embodiments, it will be understood that they are not intended to limit the invention to these embodiments. On the contrary, the invention is intended to cover alternatives, modifications and equivalents, which may be included within the spirit and scope of the invention as defined by the appended claims. Furthermore, in the following detailed description of the present invention, numerous specific details are set forth in order to provide a thorough understanding of the present invention. However, it is understood that the present invention may be practiced without these specific details. In other instances, well-known methods, procedures, components, and circuits have not been described in detail as not to unnecessarily obscure aspects of the present invention.

Referring now to Figure 3, a flow diagram of a computer-implemented method of configuring a circle of trust between a first and second affiliated entity, in accordance with one embodiment of the present invention, is shown. The method of the present embodiment may be realized as a series of instructions (e.g., code) and information (e.g., data) that reside on a computer-readable medium, such as computer memory, and are executed and manipulated by a processor. When executed, the instructions cause the processor to implement the process of configuring a circle of trust. As depicted in Figure 3, the method comprises receiving a certificate of the first affiliated entity by a second

affiliated entity, at step 310. The certificate of the first affiliated entity is stored in a trusted partner list accessable to the second affiliated entity, at step 330. The method further comprises receiving a certificate of the second affiliated entity by the first affiliated entity, at step 340. The certificate of the second affiliated entity is stored in a trusted partner list accessable to the first affiliated entity, at step 360.

The method may be repeated for each pair of affiliated entities in the network. Thus, the trusted partner list for each entity will contain a record, comprising the certificate, for each entity on the network that is affiliated with the particular entity.

In another embodiment of the present invention, the method comprises the steps 310, 330, 340 and 360 of the above-described embodiment. The method further comprises receiving a network address of the first affiliated entity by the second affiliated entity, at step 320. The network address of the first affiliated entity is stored in the trusted partner list accessable to the second affiliated entity, at step 330. The method further comprises receiving a network address of the second affiliated entity by the first affiliated entity, at step 350. The network address of the second affiliated entity is stored in the trusted partner list accessable to the first affiliated entity, at step 360.

The method may be repeated for each pair of affiliated entities in the network. Thus, the trusted partner list for each entity will contain a record, comprising the certificate and a network address, for each entity on the network that is affiliated with the particular entity.

In yet another embodiment of the present invention, the method comprises receiving a network address of the first affiliated entity by a third affiliated entity. The network address of the first affiliated entity is stored in a trusted partner list accessable to the third affiliated entity. The method further comprises receiving a network address of the third affiliated entity by the first affiliated entity. The network address of the third affiliated entity is stored in a trusted partner list accessable to the first affiliated entity.

The method may be repeated for each pair of affiliated entities in the network. Thus, the trusted partner list for each entity will contain a record, comprising the network address, for each entity on the network that is affiliated with the particular entity. Furthermore, any combination of the above-described three embodiments may be utilized. Thus, the trusted partner list of a given entity may contain a record for each entity on the network that is affiliated with the particular entity. The record of each of the trusted partners may comprise the corresponding certificate, network address, and/or the like,

Referring now to Figure 4, an exemplary trusted partner list, in accordance with one embodiment of the present invention, is shown. The trusted partner list comprises a plurality of records 410. In one implementation, each record 410 comprises an identifier of the particular trusted entity 420 and a certificate corresponding to the particular trusted entity 430. In another implementation, each record 410 comprises an identifier of the particular trusted entity 420 and a network address (e.g., an internet protocol (IP) address) 440. In yet another implementation, each record 410 comprises an identifier of the particular trusted entity 420, a certificate corresponding to the particular trusted

entity 430, and a network address 440 corresponding to the particular trusted entity. For example, the trusted partner list for Serve A may comprise identifiers for servers B and C, certificates of B and C, and network address of B and C, respectively.

Referring now to Figure 5, a block diagram of a system providing for configuring a circle of trust, in accordance with one embodiment of the present invention, is shown. As depicted in Figure 5, each affiliated entity (e.g., server A and B) 510, 520 provides for configuring the circle of trust utilizing an administration module 530, 540 and a trusted partner list 550, 560. The administration modules 530, 540 of each affiliated entity 510, 520 provide for the exchange of credentials and storage thereof in a corresponding trusted partner list 550, 560. The credential may comprise one or more certificates (e.g., cert1, cert2, cert3), one or more network addresses (e.g., ip1, ip2, ip3) and/or the like. Accordingly, a request sent from any of the network addresses and/or a request containing any one of the certificates indicated in the credential will be acceptable.

In one implementation, the administration module 530, 540 of the first affiliated entity 510 (e.g., server A) transmits a certificate of the first affiliated entity 510 to the second affiliated entity 520 (e.g., server B). The administration module 540 of the second affiliated entity 520 receives the certificate of first affiliated entity 510 and stores it as a record in the trusted partner list 560 of the second affiliated entity 520. The administration module 540 of the second affiliated entity 520 transmits the certificate of the second affiliated entity 520 to the first affiliated entity 510. The administration module 530 of the first affiliated entity 510 receives the credential of the second affiliated

entity 520 and stores it as a record in the trusted partner list 550 of the first affiliate entity 510.

Alternatively, the exchange of certificates may be performed out-of-band (e.g., manually) and then loaded into the respective trusted partner list as per individual design. For example, one entity may user some kind of certification utility to import and store the received certificate.

In another embodiment, the administration module 530 of the first affiliated entity 510 transmits a network address, such as an internet protocol (IP) address, of the first affiliated entity 510 to the second affiliated entity 520. The administration module 540 of the second affiliated entity 520 receives the network address of the first affiliated entity 510 and stores it as a record in the trusted partner list 560 of the second affiliated entity 520. The administration module 540 of the second affiliated entity 520 transmits the network address of the second affiliated entity 520 to the first affiliated entity 510. The administration module 530 of the first affiliated entity 510 receives the network address of the second affiliated entity 520 and stores it as a record in the trusted partner list 550 of the first affiliated entity 510.

Referring now to Figure 6, a flow diagram of a computer implemented method of providing a circle of trust session between a first and second affiliated entity, in accordance with one embodiment of the present invention, is shown. The method of the present embodiment may be realized as a series of instructions (e.g., code) and information (e.g., data) that reside on a computer-readable medium, such as computer

memory, and are executed and manipulated by a processor. When executed, the instructions cause the processor to implement the process of providing a circle of trust session. As depicted in Figure 6, the method begins with a user on a client device logging onto an issuing party (e.g., first affiliated entity), at step 605. The user provides a user name, password and/or the like when logging in. The issuing party authenticates the client device, at step 610. The authentication process may comprise a simple password, a personal key identifier (PKI), a hashing function or the like.

Upon authentication of the user, the issuing party returns an authentication assertion reference to the client device, at step 615. When initiating use of a resource on another device, referred to as a relying party (e.g., second affiliated entity), the client device passes the authentication assertion reference to the particular relying party, at step 620. The relying party determines the identity of the issuing party as a function of the authentication assertion reference, at step 625.

If communication is being provided over a single socket layer (SSL) HTTP protocol or the like, the relying party sends an authentication request containing the certificate of the relying party to the issuing party, 630. The issuing party looks up the certificate in a trusted partner list, at 635. If the issuing party finds a match for the certificate in the issuing party's trusted partner list, the issuing party thereby knows the identity of the relying party and that the relying party is a trusted partner.

The issuing party thereafter returns an authentication assertion to the relying party, at 655. The authentication assertion states that the client device was, in fact,

authenticated by a particular method at a specific time and the like information. The relying party then provides the client device with the requested resource without requiring the user to separately logon to the relying party, at 660.

If communication is being provided over a non-single socket layer (SSL) HTTP protocol or the like, the relying party sends an authentication request to the issuing party, at step 640. The issuing party parses the header of the communication packet to determine a network address of the relying party, at step 645. The issuing party looks up the network address in a trusted partner list, at step 650. If the issuing party finds a match for the network address in the issuing party's trusted partner list, the issuing party thereby knows the identity of the relying party and that the relying party is a trusted partner.

Again, the issuing party returns an authentication assertion to the relying party, at step 655. The authentication assertion states that the client device was, in fact, authenticated by a particular method at a specific time and the like information. The relying party then provides the client device with the requested resource without requiring the user to separately logon to the relying party, at step 660.

Referring now to Figure 7, a block diagram of a system providing for a circle of trust session, in accordance with one embodiment of the present invention, is shown. As depicted in Figure 7, the circle of trust session comprises a client device 710, an issuing party (e.g., first affiliated entity) 715 and a relying party (e.g., second affiliated entity) 740. During the circle of trust session, a user login request is passed by the client device

710 to the issuing party 715 by a SAML authentication request. The authentication request is transmitted from the client device 710 to the authentication module 725 of the issuing party 715. The authentication request, containing the user name, password and the like, is passed within a SOAP envelope via HTTP. The authentication module 725 processes the packet to retrieve the user name, password and the like.

The user name, password and the like is passed to an authentication module 725 of the issuing party 715. The authentication process provided by the authentication module 725 may comprises a simple password, a personal key identifier (PKI), a hashing function or the like. The authentication module 725 returns a SAML authentication assertion to the client device 710, if the authentication process is successful. The authentication assertion comprises an authentication assertion reference.

The client device 710 passes the authentication assertion reference to each relying party (e.g., server B) 740 when initiating use of resources thereon. The relying party 720 can determine the particular issuing party 715, which authenticated the client 710, from the authentication assertion reference. Therefore, the relying party 740 sends a SAML authentication query to the issuing party 715.

In one implementation, if the communication is provided over a single socket layer (SSL) HTTP protocol, the SAML authentication query, transmitted by the session module 745 of the rely party 740 to the session module 720 of the issuing party 715, contains the relying party's 720 certificate. Upon receipt of the authentication query, the session module 720 of the issuing party 715 looks up the certificate in its trusted partner

list 730. If a match is found in the issuing party's 715 trusted partner list 730, the trusted party 715 knows that the relying party 740 is the requesting entity and that the relying party 740 can be trusted. Therefore, the session module 720 of the issuing party 715 returns a SAML authentication assertion to the session module 745 of the relying party 740. The SAML authentication assertion states that the client device 710 was, in fact, authenticated by a particular method at a specific time and the like information.

In another implementation, if the communication is provided over a non-single socket layer (SSL) protocol, the relying party's 740 network address, such as its internet protocol (IP) address, is determined from the transmission layer protocol header, such as the HTTP header. The session module 720 of the issuing party 715 looks up the network address in its trusted partner list 730. If a match is found in the issuing party's 715 trusted partner list 730, the trusted party 715 knows that the relying party 740 is the requesting entity and that the relying party 740 can be trusted. Therefore, the session module 720 of the issuing party 715 returns a SAML authentication assertion to the session module 745 of the relying party 740. The SAML authentication assertion states that the client device 710 was, in fact, authenticated by a particular method at a specific time and the like information.

Accordingly, embodiments of the present invention provide an open and interoperable single sign-on for secure authentication and authorization. Embodiments of the present invention also provide affiliated entities a means for determining that a user has been authenticated. Embodiments of the present invention also provide the affiliated

entities a means for determining whether they can trust the authentication provided by another entity. Furthermore, embodiments of the present invention allow each affiliated entity to use their own chosen methods of authentication (e.g., personal key identifier (PKI), hash, password, or the like). Hence, the above-described systems and methods, provided by embodiments of the present invention, advantageously increase user satisfaction and productivity.

The foregoing descriptions of specific embodiments of the present invention have been presented for purposes of illustration and description. They are not intended to be exhaustive or to limit the invention to the precise forms disclosed, and obviously many modifications and variations are possible in light of the above teaching. The embodiments were chosen and described in order to best explain the principles of the invention and its practical application, to thereby enable others skilled in the art to best utilize the invention and various embodiments with various modifications as are suited to the particular use contemplated. It is intended that the scope of the invention be defined by the Claims appended hereto and their equivalents.